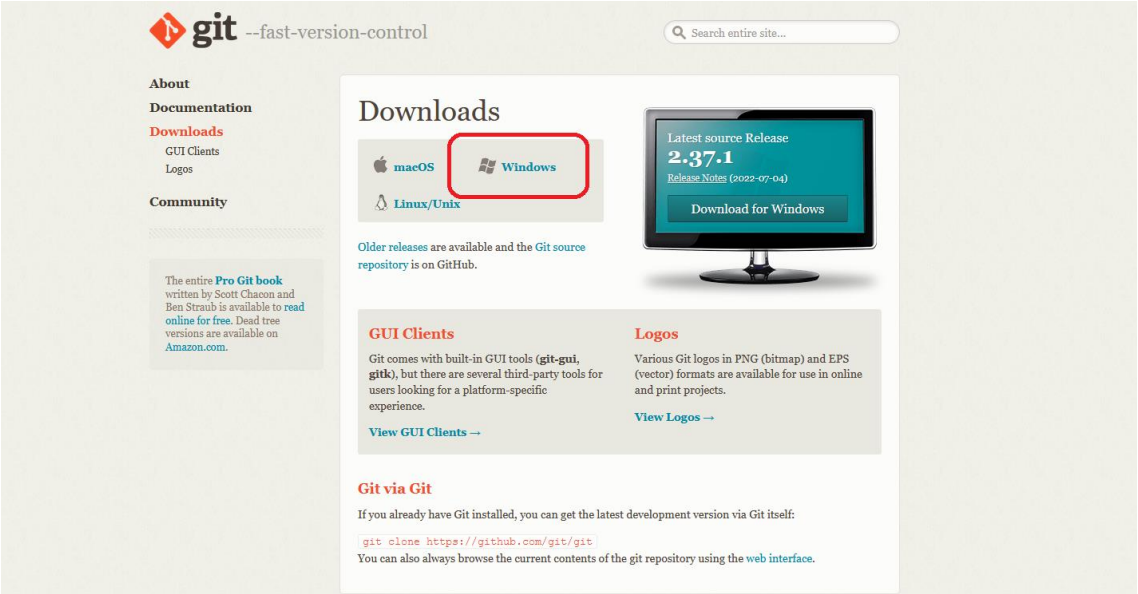
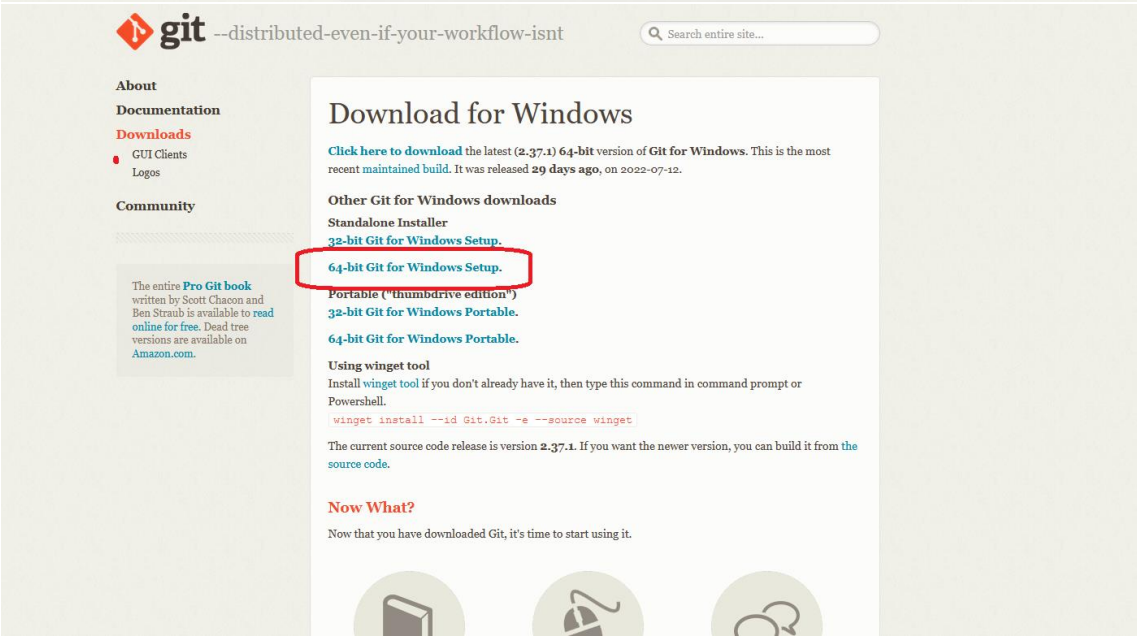
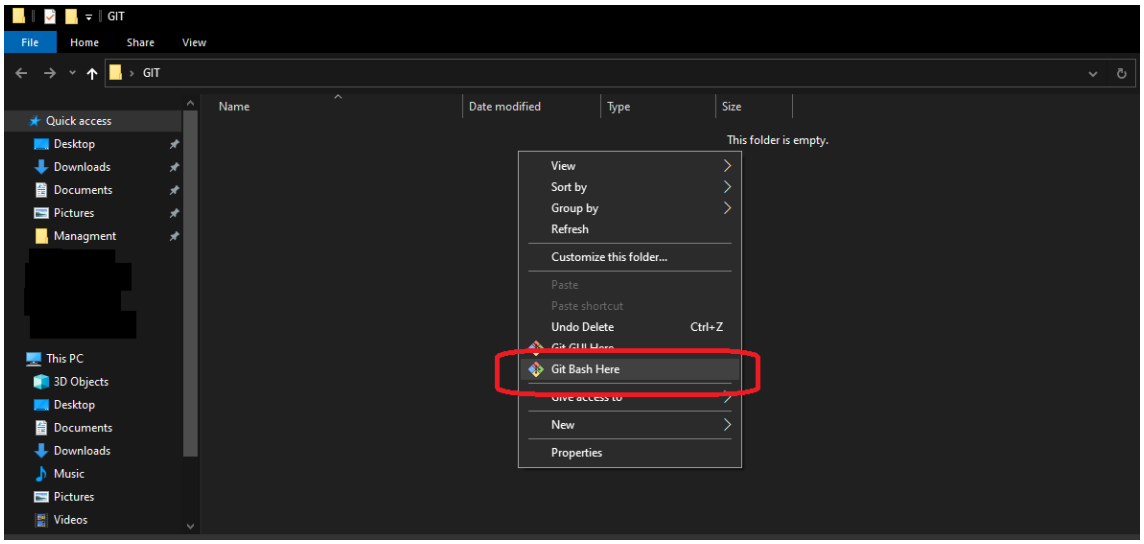
Git Bash

- Faça download no link: https://git-scm.com/downloads





- Instale o GIT.
- Na pasta em que você deseja realizar o download do repositório criado, abra o Git Bash .



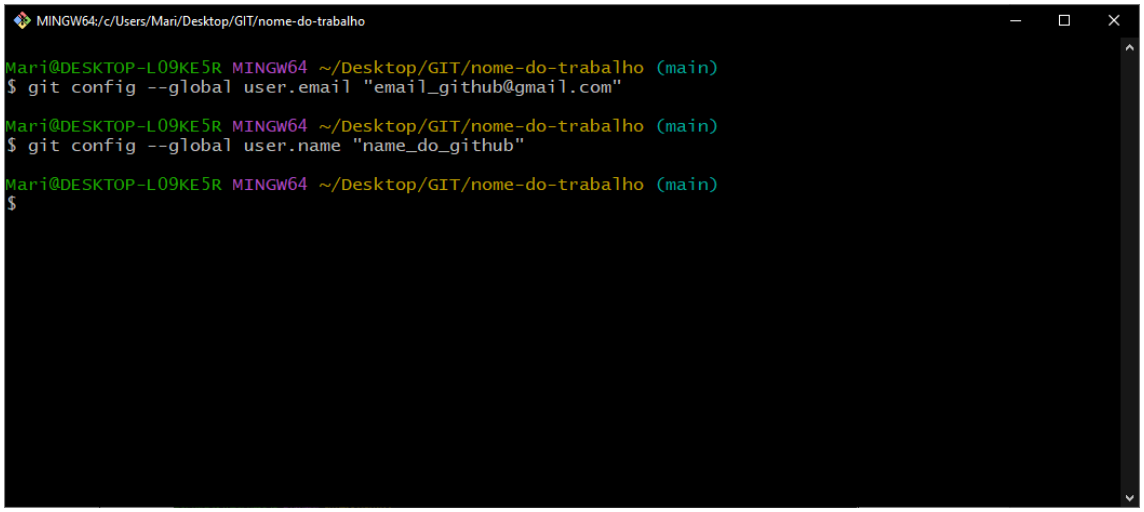
Vamos configurar suas credenciais:

Use commando:

`git config - -global user.email "email_github@gmail.com"`

Em seguida use o comando (O seu user name é visível na sua conta do GitHub):

`git config - -global user.name "name_do_github"`



Vamos abrir um parêntese aqui:

Os seguintes comandos são importantes:

- `ls` : quando você pretender lista as pasta ou arquivos.
- cd nome_pasta/ : quando você pretender entrar em uma pasta específica.
- `cd ..` : volta uma pasta.

Exemplo:

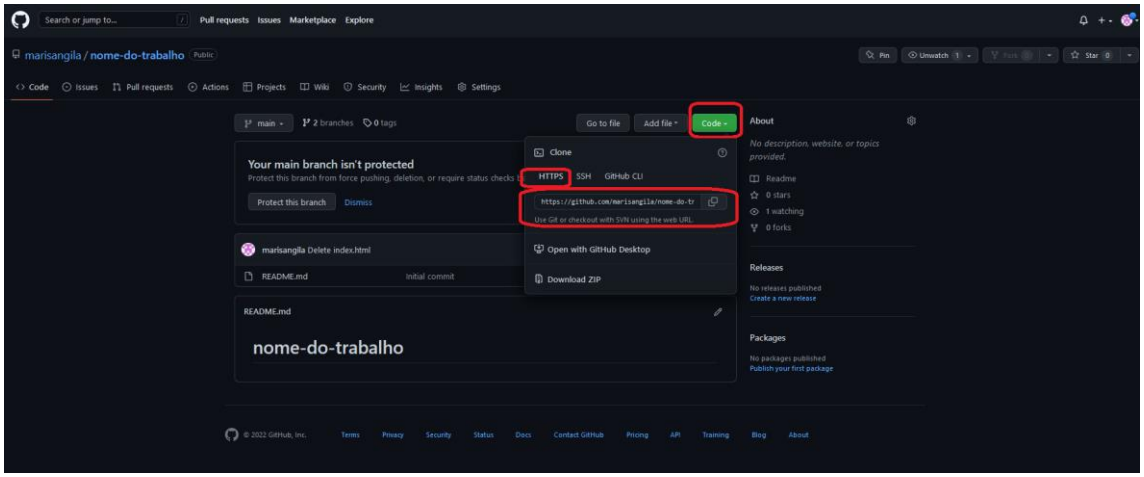
```
MINGW64:/c/Users/Mari/Desktop/GIT

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT
$ ls
nome-do-trabalho/  pasta1/  pasta2/

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT
$ cd pasta1

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT/pasta1
$ ls
pasta3/

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT/pasta1
$ cd pasta3/

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT/pasta1/pasta3
$ ls
texto.txt

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT/pasta1/pasta3
$ cd ..

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT/pasta1
$ cd ..

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT
$ ls
nome-do-trabalho/  pasta1/  pasta2/

Mari@DESKTOP-L09KE5R MINGW64 ~/Desktop/GIT
$
```

Estamos dentro da pasta GIT.
Primeiro, listamos o que há dentro da pasta GIT.
Temos a pasta nome-do-trabalho, pasta1 e pasta2.
Depois a pasta1 foi acessada.
Dentro da pasta1 listamos o que há dentro dela, encontramos a pasta3.
A pasta3 foi acessada.
Dentro da pasta3 listamos o que há dentro dela, encontramos um arquivo de texto.
Voltamos da pasta3 para a pasta1
Voltamos da pasta1 para a pasta inicial (GIT)
Listamos novamente, para visualizar o que há na pasta GIT

Seguindo:

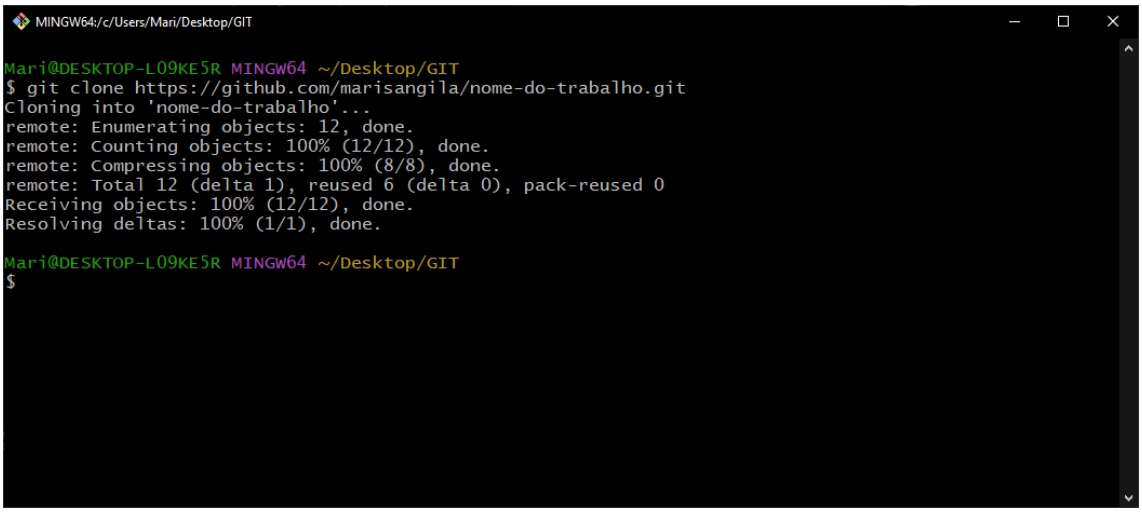
Agora vamos **clonar** seu repositório no GitHub para o computador:

- Copie a URL do seu repositório

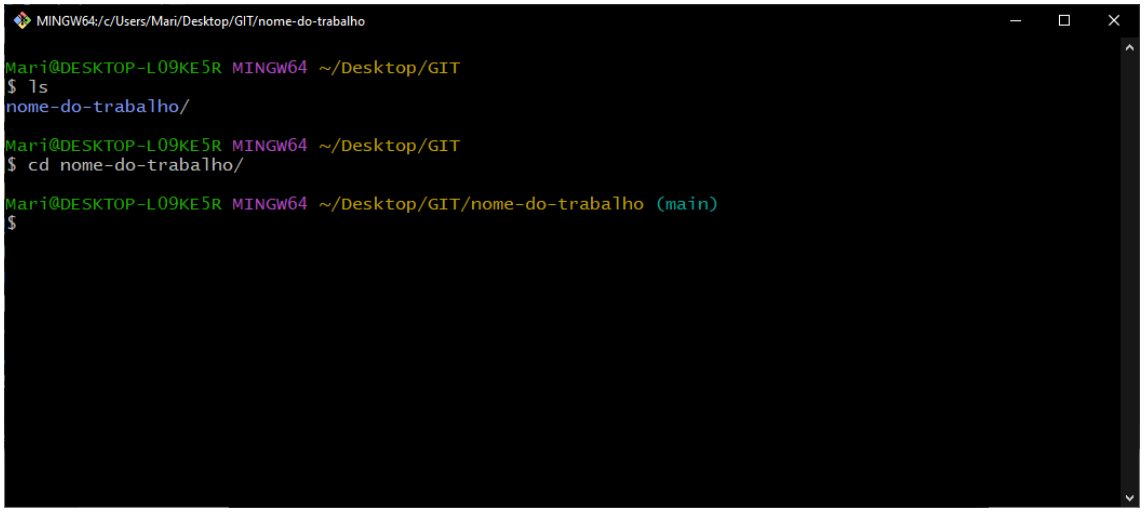


- Agora novamente no Git Bash use o comando:

`git clone "link_que_voce_copiou"`

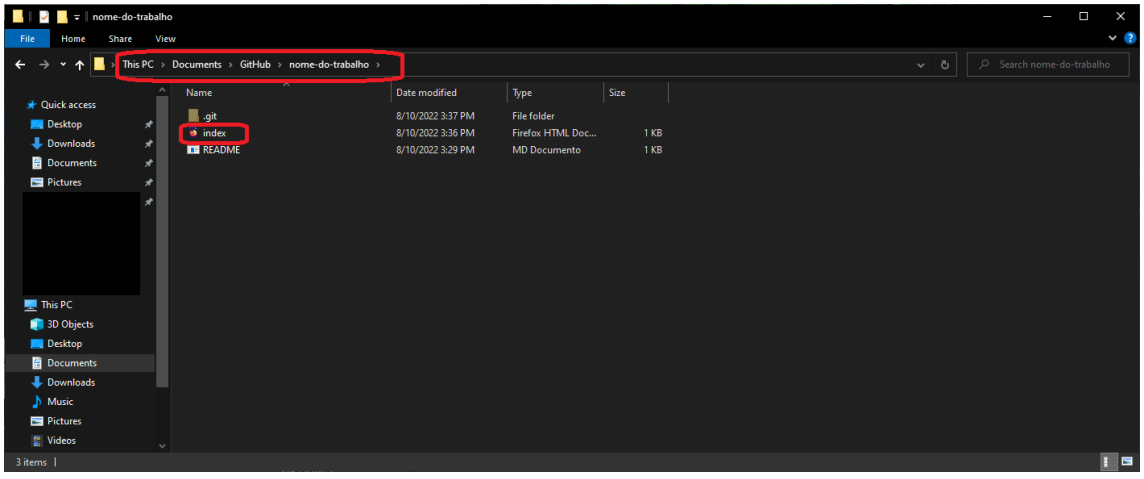


- Entre na pasta:



Vamos criar um arquivo de teste

- Crie um arquivo qualquer dentro da pasta do seu repositório:

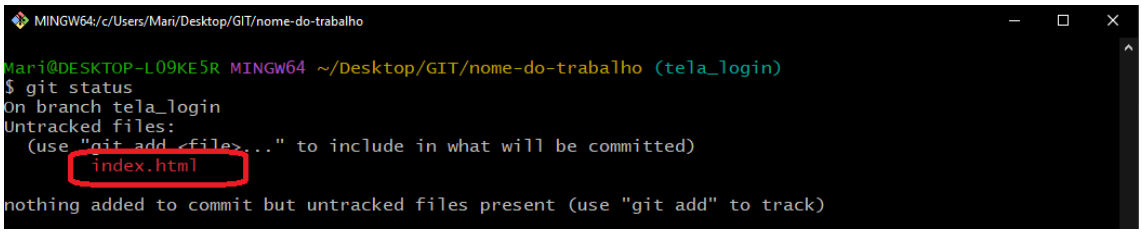


- Agora use o comando:

`git status`

O comando **git status** deve ser usado sempre que você precisar chegar algo (Exemplo: quais arquivos foram editados).

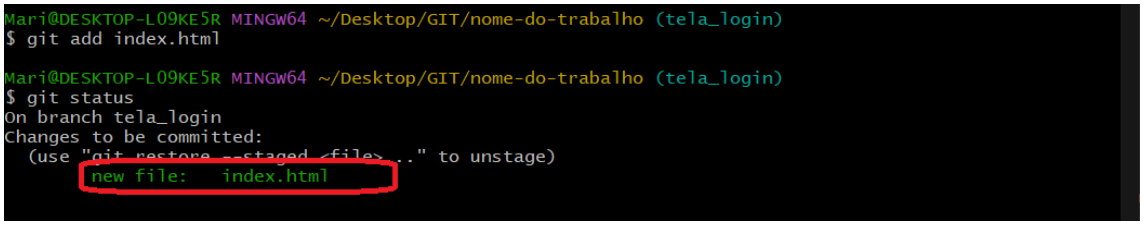
O novo arquivo aparecerá em como **untracked files** (vermelho):



- Adicione o arquivo a **área de staging** usando o comando:

`git add "nome_arquivo.html"`

Se você usar o comando **git status** novamente verá que o arquivo como **new file** (verde):

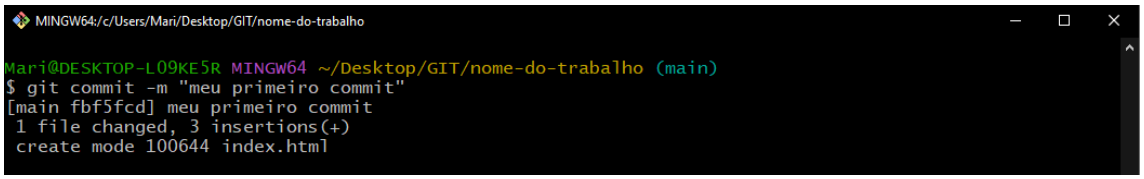


Agora é necessário realizar um **commit** usando o comando:

`git commit -m "meu primeiro commit"`

No **commit** você define quais arquivos ou alterações serão enviadas ao GitHub, além disso, você pode definir uma mensagem.

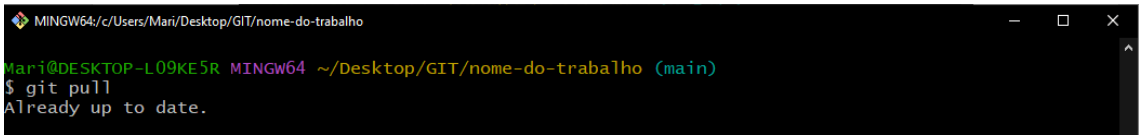
(Lembre-se usar aqui uma mensagem sinalizando qual alteração você fez, assim fica fácil mapear alterações futuramente).



Antes de enviar suas alterações ao GitHub é necessário garantir que o os arquivos na sua máquina estão com as últimas alterações existentes no repositório do GitHub (Outro integrante da equipe pode ter realizado alguma alteração).

- Para isso use o comando:

`git pull`



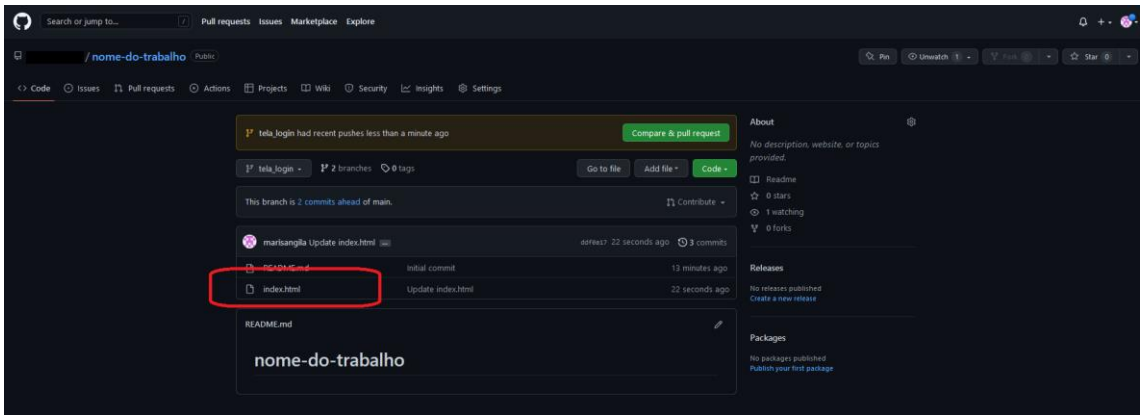
Caso tenha atualizações, o git irá realizar download, caso contrário aparecerá a mensagem "**Already up to date**"

Finalmente, é possível enviar suas alterações para o repositório no GitHub, assim os outros integrantes da equipe poderão ver suas alterações.

- Para isso use o comando:

`git push`

Agora, suas alterações apareceram no GitHub!



Agora os outros integrantes podem ver suas alterações e transferir essas alterações para o computador usando **GIT PULL**!